

Iterative Learning Control Based on Stretch and Compression Mapping for Trajectory Tracking in Human-robot Collaboration

1st Jingkang Xia

*School of Electrical Engineering
Southwest Jiaotong University
Chengdu, China
xia-jingkang@my.swjtu.edu.cn*

2nd Deqing Huang

*School of Electrical Engineering
Southwest Jiaotong University
Chengdu, China
elehd@my.swjtu.edu.cn*

3rd Yanan Li

*Department of Engineering and Design
University of Sussex
Brighton, UK
yl557@sussex.ac.uk*

4th Junpei Zhong

*School of Science and Technology
Nottingham Trent University
Nottingham, UK
joni.zhong@ntu.ac.uk*

Abstract—In this paper, a novel iterative learning control (ILC) scheme based on stretch and compression mapping is proposed for a robotic manipulator to learn a trajectory that is planned by a human partner but unknown to itself, which is a typical task in many human-robot interaction applications. The proposed method updates the robot's reference trajectory in an iterative learning manner to minimize the interaction force between the robot and the human partner. Thus, the robot can track the human partner's repetitive trajectory with a small interaction force, leading to little control effort from the human. As the human is involved in the control loop, there are various uncertainties in the system, including variable iteration period in the task under study. The stretch and compression mapping is proposed to solve this problem. Simulations on a robotic manipulator are carried out to show the effectiveness of the proposed method in human-robot collaboration. Results also illustrate better performance of the proposed ILC compared to other ILC methods with variable periods.

Index Terms—Human-robot interaction, Robotic control, Iterative learning control.

I. INTRODUCTION

Robots are starting to be more used in daily applications with the development of the industrialization and information technology, and there is an increasing demand for research in control of robots in health care, assistive devices and other applications of human-robot interaction (HRI) [1]–[4]. How to design a simple, effective and safe control scheme has been a focus in the field of HRI, especially in the subfield of physical human-robot interaction (pHRI) [5]. In many pHRI tasks, robots are required to learn the human partners' movement intentions, e.g. carrying, polishing and welding [6].

Iterative learning control (ILC) is an effective control scheme for improving performance of uncertain dynamic systems that follow the same operation where a task is repeatedly completed in a finite time interval [7]. By dealing with repetitive uncertainties and external disturbances, ILC has been widely used in control of robots. For example, [8] proposes an adaptive iterative learning control (AILC) for robotic system with both structured and unstructured uncertainty. [9] designs an ILC for trajectory tracking of rigid robot manipulators subject to external disturbances and performing repetitive tasks. [10] further studies trajectory tracking control of rigid robot manipulators with model uncertainty by ILC.

Although ILC has shown excellent capability in dealing with system uncertainty and repetitive interference [11], [12], conventional ILC has a critical assumption of a fixed time interval in different iterations. However, in many practical tasks, especially those with human partners involved, the iteration period may vary in different iterations. For instance, iterative learning or repetitive learning can be used to control the gait tracking of an exoskeleton robot [13]. Since the speed of the human user in each iteration is inconsistent, the iteration length may be variable so the traditional ILC is no longer applicable. In addition, due to data collection errors, there may be inconsistency in the time length of data collection in each iteration, and the missing data or extra data will lead to the traditional ILC unreliable [14]. This paper will discuss a repetitive trajectory tracking task in space, where the robot needs to track a fixed trajectory planned by a human partner but unknown to it beforehand. Due to the uncertainty of the human behavior, the period of tracking the desired trajectory will change in each iteration, so that the ILC with fixed iteration period will lose its capability to ensure the tracking performance.

*This work is supported in part by the Sichuan Science and Technology Program under Grants 2019YFG0345 and 2019YJ0210 and National Natural Science Foundation (NNSF) of China under Grants 51805025, 61773323 and 61733015. (Corresponding author : Yanan Li.)

Based on the above discussions, this paper develops a period-varying ILC that ensures learning convergence in the presence of human uncertainty. In the literature of ILC, some relevant works [14]–[18] investigate the problem of period-varying which lay the foundation of our method. In particular, [14] introduces an ILC scheme based on an iteratively moving average operator for nonlinear dynamic systems with randomly varying iteration periods. [17] proposes an ILC scheme with an iteration-average operator for discrete-time linear systems where the iteration period could be randomly varying. Different from the above two methods, [18] adopts an adaptive ILC framework for continuous-time parametric nonlinear systems with varying iteration periods. Similar to other ILC methods as discussed above, these three methods require a predefined desired output trajectory. In this paper, the human’s unknown desired trajectory will be first learned by using a human limb model and by formulating a zero force regulation problem. Then, it will be tracked by the robot so it can reduce human’s control effort. Compared to related works in the literature, this paper includes the following contributions:

- 1) A novel iterative learning method is proposed for nonlinear systems with randomly varying iteration lengths varying, with the introduction of an stretch and compression mapping.
- 2) The proposed ILC scheme is applied to learn a desired trajectory that is unknown to the robot but planned by its human partner, so that interaction force will gradually decrease and the human partner’s control effort will be reduced.
- 3) Comparison of the existing methods and the proposed one using simulations on a robotic manipulator is carried out to demonstrate the advantages of the proposed ILC in human-robot collaboration.

The rest of the paper is organized as follows. Section II introduces the main problem formulation. In Section III, a non-standard ILC is proposed to update the reference trajectory of the robotic end-effector. The properties of the proposed scheme are demonstrated by comparative simulations on a robotic manipulator in Section IV. Conclusions are given in Section V.

II. PROBLEM FORMULATION

A. System Description

In this paper, we consider a typical human-robot collaboration scenario that is composed of a robotic manipulator and its human partner as shown in Fig. 1. The human partner guides a robotic manipulator to learn a task trajectory, e.g. carrying an object starting from an initial position to a target position with human partner’s desired trajectory. This trajectory is determined by the human partner but cannot be preprogrammed to the robotic manipulator. The interaction force between the human partner’s hand and the robotic manipulator is measured by a force sensor at the end-effector of the robotic manipulator.

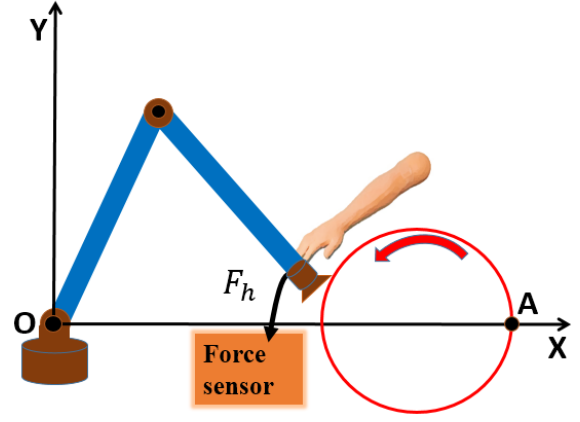


Fig. 1. Human-robot collaboration scenario

The dynamics of this n -degrees-of-freedom (n -DOF) robotic manipulator can be described as

$$H(q(t))\ddot{q}(t) + C[q(t), \dot{q}(t)]\dot{q}(t) + G(q(t)) = u(t) + J^T(t)F_h(t) \quad (1)$$

where $q(t), \dot{q}(t), \ddot{q}(t) \in \mathbb{R}^n$ represent the manipulator’s joint position, velocity and acceleration vectors, respectively; $H(q(t)) \in \mathbb{R}^{n \times n}$ is the symmetric positive definite mass matrix; $C[q(t), \dot{q}(t)]\dot{q}(t) \in \mathbb{R}^n$, $G(q(t)) \in \mathbb{R}^n$ denote the torques due to centrifugal force and gravity; $u(t) \in \mathbb{R}^n$ is the joint torque applied by the manipulator’s actuators; $J(t) \in \mathbb{R}^{n \times n}$ is the Jacobian matrix that relates the joint velocity to the linear and angular velocities of the end-effector and $F_h(t) \in \mathbb{R}^n$ is the robot/human interaction force that can be measured by a force sensor.

It is often desirable to describe the manipulator dynamics in the Cartesian space for the convenience of analysis, when the interaction takes place at the end-effector. The robot dynamics in the Cartesian space are given by

$$H(t)\ddot{X}(t) + C(t)\dot{X}(t) + G(t) = J^{-T}(t)u(t) + F_h(t) \quad (2)$$

where $X(t) \in \mathbb{R}^n$ represents the position of the end-effector and $\dot{X}(t) = J(t)\dot{q}(t)$, $\ddot{X}(t) = \dot{J}(t)\dot{q}(t) + J(t)\ddot{q}(t)$, so it can be obtained that

$$\begin{aligned} H(t) &= J^{-T}(t)H(q(t))J^{-1}(t), \\ C(t) &= J^{-T}(t)[C(q(t), \dot{q}(t)) - H(q(t))\dot{J}(t)J^{-1}(t)]J^{-1}(t) \\ G(t) &= J^{-T}(t)G(q(t)) \end{aligned} \quad (3)$$

Let $e(t) = X(t) - X_r(t)$, where $X_r(t)$ is the reference trajectory of the manipulator’s end-effector, and $e(t)$ is the tracking error vector at time t . Thus, $X(t) = X_r(t) + e(t)$, $\dot{X}(t) = \dot{X}_r(t) + \dot{e}(t)$, $\ddot{X}(t) = \ddot{X}_r(t) + \ddot{e}(t)$, combined with Eq. (2), the tracking error dynamics can be described as follows

$$H_e(t)\ddot{e}(t) + C_e(t)\dot{e}(t) + G_e(t) = F(t) + F_h(t) \quad (4)$$

where $H_e(t) = H_x(t)$, $C_e(t) = C_x(t)$, $G_e(t) = H_x(t)\ddot{X}_r(t) + C_x(t)\dot{X}_r(t) + G_x(t)$ and $F(t) = J^{-T}(t)u(t)$.

According to [19], dynamics of a robot's environment can be described by a spring model. This corresponds to the equilibrium point control model that describes the human motor control [20], which is given below

$$F_h(t) = -K_h(X(t) - X_h(t)) \quad (5)$$

where K_h is the equivalent stiffness matrix and $X_h(t)$ is the desired trajectory of the human. With Eqs. (4) and (5), we have a full description of the system dynamics and are ready to discuss the design objective of the robot's controller.

B. Design objective

We design the robot's controller

$$F(t) = H_e(t)\ddot{e}(t) - K_v\dot{e}(t) - K_p e(t) + C_e(t)\dot{e}(t) + G_e(t) \quad (6)$$

where K_v , K_p are two diagonal matrices. Note that we consider a two-dimensional case for analysis simplicity but the proposed method can be extended to a higher dimension.

By combining this controller with Eq. (4), a desired impedance model is obtained

$$K_v\dot{e}(t) + K_p e(t) = F_h(t) \quad (7)$$

From (5) we get $F_h(t) = -K_h(X(t) - X_h(t)) = -K_h[X(t) - X_r(t) + X_r(t) - X_h(t)]$. By defining $e'(t) = X_h(t) - X_r(t)$, Eq. (7) can be rewritten as

$$K_v\dot{e}(t) + K_p e(t) = -K_h(e(t) - e'(t)) \quad (8)$$

Let $\hat{K}_p = K_p + K_h$, then Eq. (8) becomes

$$K_v\dot{e}(t) + \hat{K}_p e(t) = K_h e'(t) \quad (9)$$

Taking the Laplace transform of Eqs. (7) and (9), we can get

$$(K_v s + K_p)e(s) = F_h(s) \quad (10)$$

$$K_v s e(s) + \hat{K}_p e(s) = K_h e'(s) \quad (11)$$

By defining $F'_h(s) = \frac{F_h(s)}{K_v s + K_p}$, Eq. (11) can be rewritten as

$$K_v s F'_h(s) + \hat{K}_p F'_h(s) = K_h e'(s) \quad (12)$$

so the dynamics in time domain is the following form

$$K_v \dot{F}'_h(t) + \hat{K}_p F'_h(t) = K_h e'(t) \quad (13)$$

We further rewrite Eq. (13) in the following state-space form

$$\begin{cases} \dot{F}'_h(t) = A F'_h(t) + B e'(t) \\ y(t) = F'_h(t) \end{cases} \quad (14)$$

where $A = -\hat{K}_p K_v^{-1}$, $B = K_h K_v^{-1}$.

By recalling the task of human-robot collaboration, we aim to achieve $e'(t) = 0$, i.e. to make the robot's reference trajectory identical to the human partner's desired trajectory. This leads to $F'_h(t) = 0$ according to Eq. (14). Therefore, for the system in Eq. (14), we consider its desired input as $e'_d(t) = 0$ and desired output as $y_d(t) = 0$. In the following section, we will introduce an ILC to achieve such a desired

system by considering a repetitive tracking task, i.e. the human partner repeats the same movement within a time period to guide the robot along their desired trajectory. It is noted that this period is likely to be inconsistent due to uncertainties of human behavior, e.g. it is difficult for the human partner to repeat the movement with the same speed in different iterations. Therefore, the ILC scheme to be developed has to deal with this problem.

III. ILC DESIGN

Consider a class of nonlinear iterative dynamic systems

$$\begin{cases} \dot{x}_i(t) = f(x_i(t), t) + B u_i(t) \\ y_i(t) = C x_i(t) \end{cases} \quad (15)$$

where $t \in [0, T_i]$ and $i \in \mathbb{N}$ are time and iteration number, respectively, with \mathbb{N} an integer and T_i the time period of the i th iteration. $u_i(t) \in \mathbb{R}^n$, $x_i(t) \in \mathbb{R}^m$, $y_i(t) \in \mathbb{R}^l$ denote input, state and output of the system. B and C are the constant matrices with appropriate dimensions. It is noted that $f(x, t)$ is Lipschitz continuous with respect to x , so that there is $M > 0$ for $\|f(x_1, t) - f(x_2, t)\| \leq M \|x_1 - x_2\|$.

For the above system, there is a unique control input $u_d(t) \in \mathbb{R}$ that allows the system to achieve the desired output trajectory $y_d(t) \in \mathbb{R}$, $t \in [0, T_d]$, such that

$$\begin{cases} \dot{x}_d(t) = f(x_d(t), t) + B u_d(t) \\ y_d(t) = C x_d(t) \end{cases} \quad (16)$$

where $x_d(t) \in \mathbb{R}^m$ is the desired state.

If the actual iteration period is equal to the desired one as $T_i = T_d$, the output error $e_i(t)$ can be defined as

$$e_i(t) = y_d(t) - y_i(t) \quad (17)$$

A simple and effective open-loop D-type learning law can be designed as

$$u_{i+1}(t) = u_i(t) + \tau \dot{e}_i(t) \quad (18)$$

which is a conventional iterative learning method. If $T_i \neq T_d$, there are two cases $T_i < T_d$ and $T_i > T_d$ where the above learning law is not applicable. In particular, when $T_i < T_d$ the system outputs are missing after T_i and when $T_i > T_d$ there are more system outputs than desired ones, so we cannot compute the tracking error $e_i(t)$ required in the learning law. To address this problem, the learning law (18) has to be redesigned.

[14] introduces a method to deal with varying period. The main idea is to use 0 to fill missing $e_i(t)$ if $T_i < T_d$ and remove abundant $e_i(t)$ if $T_i > T_d$. Also proved to be valid, this method has a disadvantage of slow learning, as the use of historical data is not maximized when $T_i > T_d$. In this paper, we introduce a stretch and compression operator s to ensure that the data for each iteration is fully utilized and does not supplement irrelevant data shown as Fig. 2. s is described as follows

$$s = \frac{T_d}{T_i} t, \quad t \in [0, T_i], \quad s \in [0, T_d] \quad (19)$$

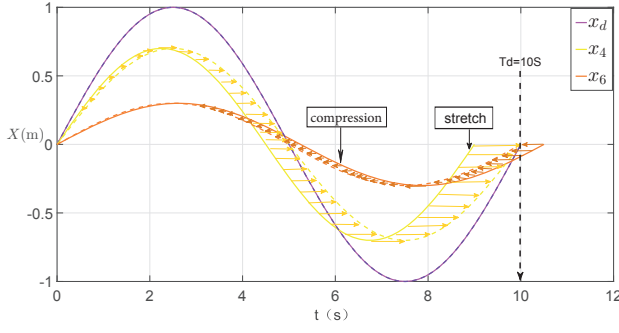


Fig. 2. An illustration of the proposed method: if $T_i < T_d$, the stretch operator s is used to extend the time domain; if $T_i > T_d$, the compression operator s is used to compress the time domain. Thus, the signal $e_i(s)$ on the time interval $[0, T_d]$ can be obtained.

By using this operator s , we establish a mapping from $y_d(t)$ and $y_i(t)$ to $y_d(s)$ and $y_i(s)$, respectively. Then the system can be analyzed in the s domain, where the new time period in different iterations becomes the same.

Define a modified tracking error

$$e_i^*(s) = y_d(s) - y_i(s) \quad (20)$$

Taking the derivative of both sides of Eq. (5) with respect to s , we have

$$\dot{e}_i^*(s) = \dot{y}_d(s) - \dot{y}_i(s) \quad (21)$$

In order to improve the stability and fault tolerance of the system and reduce the influence of system noise, an iteratively average-operator is introduced to use historical information, which is described as

$$\Phi\{u_i(s)\} = \frac{1}{m+1} \sum_{j=0}^m u_{i-j}(s) \quad (22)$$

for a sequence $u_{i-m}(s)$, $u_{i-m+1}(s)$, $u_{i-m+2}(s)$..., $u_i(s)$ with $m \geq 1$, including the recent $m+1$ iterations. The new ILC learning law is given as follows

$$u_{i+1}(s) = \Phi\{u_i(s)\} + \sum_{j=0}^m \beta_j \dot{e}_{i-j}^*(s), \quad s \in [0, T_d] \quad (23)$$

where the learning rates $\beta_j \in \mathbb{R}^+$, $j = 0, 1, 2, \dots$. When $u_{i+1}(s)$ is obtained, it can be converted back to the time domain and we get $u_{i+1}(t)$ that is the control input in the next iteration.

As long as the control system in the s domain is convergent, the convergence of the system in the time domain can be guaranteed. There are three conditions for convergence:

- $f(x, s)$ is Lipschitz continuous with respect to x , so that there is $M_s > 0$ for $\|f(x_1, s) - f(x_2, s)\| \leq M_s \|x_1 - x_2\|$.
- Each iteration starts at the same position $x_i(0) = x_d(0)$.
- There is a unique desired output $y_d(s)$ resulting from the desired input $x_d(s)$.

More specifically, when initial control input $u_0(s)$ and initial state $x_i(0)$ are given, the necessary and sufficient condition

for sequences $x_s(s)$, $y_i(s)$ and $u_i(s)$ to uniformly converge to $x_d(s)$, $y_d(s)$ and $u_d(s)$, respectively, is described as

$$\sum_{j=0}^m \left\| \frac{I}{m+1} - \beta_j CB \right\| < 1 \quad (24)$$

The following control objectives are achieved: $\lim_{i \rightarrow \infty} \delta u_i(s) = 0$, $\lim_{i \rightarrow \infty} \delta x_i(s) = 0$ and $\lim_{i \rightarrow \infty} e_i(s) = 0$, where

$$\begin{cases} \delta u_i(s) = u_d(s) - u_i(s) \\ \delta x_i(s) = x_d(s) - x_i(s) \\ e_i(s) = y_d(s) - y_i(s) \end{cases} \quad (25)$$

According to Eq. (23), the ILC learning law for the robot can be designed as

$$e'_{(i+1)}(s) = \Phi\{e'_{(i)}(s)\} + \sum_{j=0}^m \beta_j \dot{F}'_{h(i-j)}(s) \quad (26)$$

Since $e'(s) = X_r(s) - X_h(s)$ where $X_h(s)$ is the human partner's same desired trajectory in different iterations, the above learning law can be used to compute the robot's reference trajectory as

$$X_{r(i+1)}(s) = \Phi\{X_{r(i)}(s)\} + \sum_{j=0}^m \beta_j \dot{F}'_{h(i-j)}(s) \quad (27)$$

Finally, $X_{r(i+1)}(t)$ is obtained by mapping $X_{r(i+1)}(s)$ from the s domain back to the time domain.

IV. SIMULATION

A. Simulation I

In order to show the effectiveness of the proposed ILC scheme and verify the convergence analysis, a nonlinear dynamical system is given as (1) where $f(x_i(t), t) = 0.5 \sin(x_i(t)) + 0.8 \cos(x_i(t))$, $x_i(0) = 0$. The desired trajectory is $y_d(t) = \sin(0.5\pi t) + \sin(\frac{2\pi t}{7})$ with the iteration period as $T_d = 10s$. The initial input signal can be set as $u_0(t) = 0$, $t \in [0, 10]s$ in the first iteration. In practice, each iteration period T_i could be directly measured, but in simulations, we assume that T_i satisfies Gaussian distribution with mean of 10s and standard deviation of 0.5s. Furthermore, we set $m = 4$, $B = 12$, $C = 0.05$ and choose $\beta_0 = 0.2$, $\beta_1 = 0.18$, $\beta_2 = 0.16$, $\beta_3 = 0.14$, $\beta_4 = 0.12$. The first simulation is mainly divided into the following two parts.

1) *Existing method*: The existing method in [14] is tested in this part for comparison. As shown in Figs. 3(a) and 3(b), different iterations are represented by curves in different colors and the end of each iteration is marked with a dot. The tracking performance is shown in Fig. 3(a), where the actual trajectory y_i approximately coincides with the reference trajectory y_d after 45 iterations. Furthermore, the tracking errors are shown in Fig. 3(b), which decrease to zero when the iteration number increases. From these results, it can be found that this method is feasible in dealing with varying time period. However, it will be demonstrated in the second part that the convergence of this learning method is relatively long compared to the proposed method.

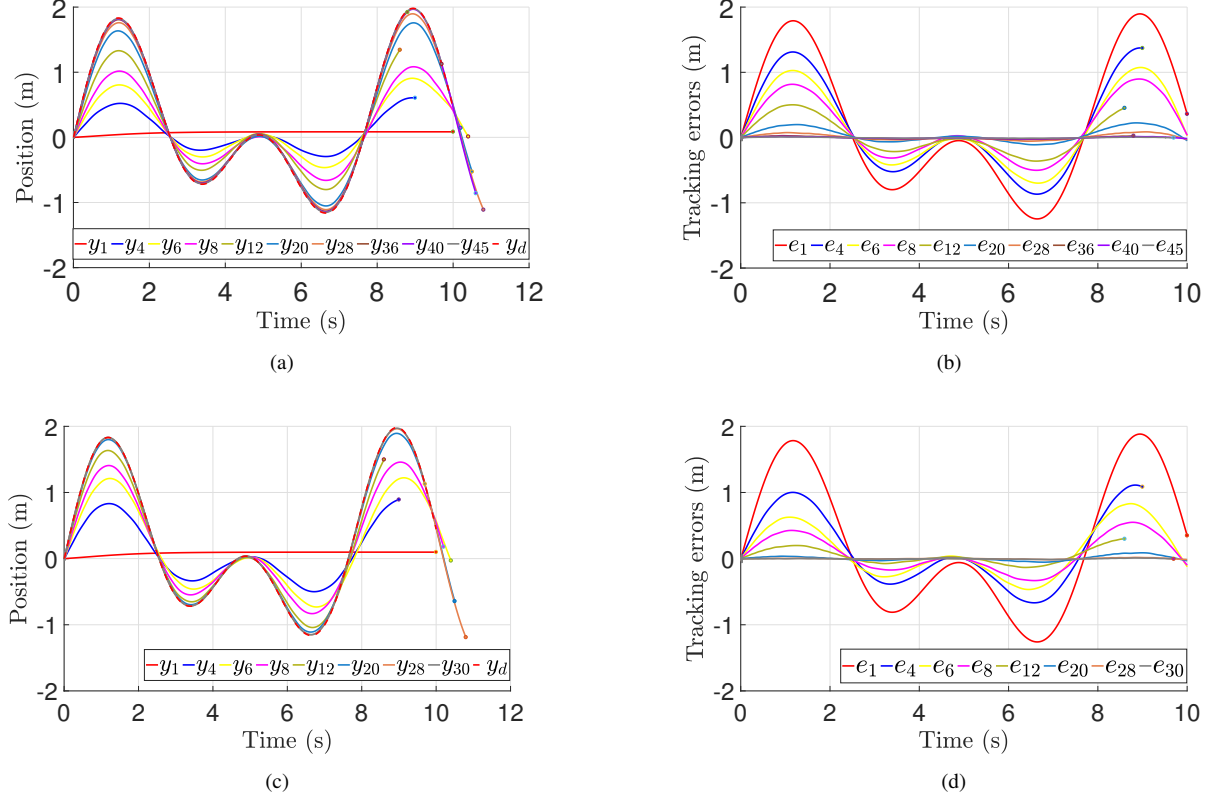


Fig. 3. (a) Trajectory tracking with varying iteration period under the existing method in [14]. (b) Trajectory tracking errors with varying iteration period under the existing method in [14]. (c) Trajectory tracking with varying iteration period under the proposed method. (d) Trajectory tracking errors with varying iteration period under the proposed method.

TABLE I
LEARNING PERIODS (S)

T_2	T_4	T_6	T_8	T_{10}	T_{12}	T_{14}	T_{16}	T_{18}	T_{20}
9.5	9.0	10.4	10.2	9.6	8.6	11.0	11.2	9.6	10.5

2) *Proposed method*: This section presents the simulation results under the proposed method in dealing with the varying iteration period problem. The parameters and iteration periods are the same as in the above section. As shown in Figs. 3(c) and 3(d), we only need 28 iterations to achieve good tracking performance and the tracking errors quickly reduce to zero. Compared with the existing method, the proposed method has an advantage of faster convergence.

B. Simulation II

In order to prove the validity of the proposed method for human-robot collaboration, we consider the scenario as introduced in Fig. 1, where a robotic manipulator and a human arm move in the $X-Y$ plane. We first set that the actual trajectory deviates from the human partner's desired trajectory, so robot needs to reduce the tracking error through collaboration with the human partner. The interaction force applied by the human partner is the key point of the proposed method, which is

used to update the robot's reference trajectory until the actual trajectory is close to the human partner's desired trajectory.

The human partner's desired trajectory is set as $X_d = \begin{bmatrix} 0.2\cos(0.2\pi t) + 0.4 \\ 0.2\sin(0.2\pi t) \end{bmatrix} m$ and the desired period is $T_d = 10s$. The robot's reference trajectory in the first iteration is $X_{r(1)} = \begin{bmatrix} 0.1\cos(0.2\pi t) + 0.5 \\ 0.12\sin(0.2\pi t) \end{bmatrix} m$ and its period is $T = 10s$. The initial position of the robotic manipulator's end-effector is set as $\mathbf{A} = \begin{bmatrix} 0.6 \\ 0 \end{bmatrix} m$ during each iteration. The control and learning parameters for simulation are $K_h = \text{diag}[0.12, 0.14]$, $K_p = \text{diag}[20, 30]$, $K_v = \text{diag}[6.8, 7.5]$, $\beta_0 = 0.25$, $\beta_1 = 0.2$, $\beta_2 = 0.18$, $\beta_3 = 0.16$ and $\beta_4 = 0.14$. Due to the involvement of human-robot interaction, there is an uncertainty in human operations, so the time period of each iteration is different. The period of each iteration is set as a value around $T = 10s$, and the value of each iteration period is given in TABLE I.

The parameters of the robotic manipulator are designed as follows. m_i , l_i , l_{ci} , I_i , $i=1, 2$ represent the mass, the length, the distance from the joint to the center of the mass and the moment of inertia respectively. We set $m_1=m_2=2kg$, $l_1=l_2=2m$, $I_1=I_2=1.0kgm^2$, and $l_{c1}=l_{c2}=1m$.

Simulation results are shown in Figs. 4 and 5, where the desired trajectory is indicated by a red dotted line. Fig. 4 shows the evolvement of trajectory tracking, which indicates that the

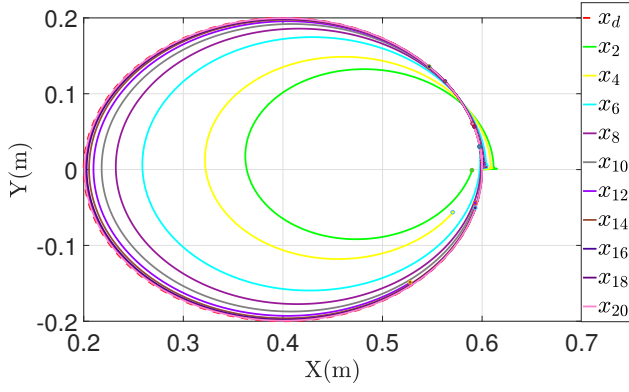


Fig. 4. Trajectory tracking in human-robot collaboration.

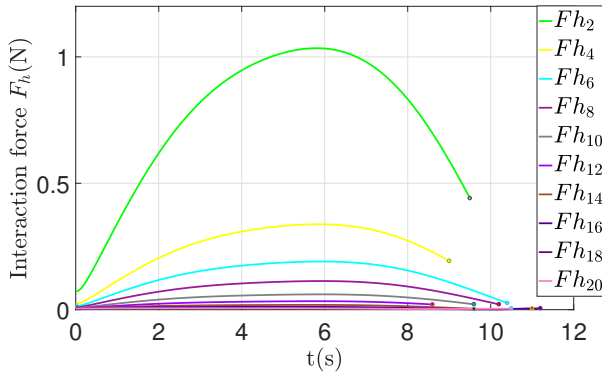


Fig. 5. Interaction force in human-robot collaboration.

actual trajectory can accurately track the desired trajectory X_d after 20 iterations. Finally, through Fig. 5, we find that with the increase of the number of iterations, the interaction force gradually decreases. When the actual trajectory is close to the human partner's desired trajectory, the interaction force is reduced to zero so the robot proactively completes the task with least human control effort.

V. CONCLUSION

In this paper, a novel iterative learning control scheme is proposed for a robotic manipulator to learn an unknown trajectory through human-robot collaboration. To achieve the design objective with varying iteration periods, a learning law is developed based on stretch and compression mapping. Compared with the existing method, the proposed one has an advantage of faster convergence. In addition, the proposed method updates the robotic manipulator's reference trajectory in an iterative learning manner to minimize the interaction force between the robot and its human partner. Thus, the robot can track the human partner's desired trajectory with repetitive interactive learning. Simulation on a 2-DOF manipulator shows the effectiveness of the proposed method in human-robot collaboration. These results show that the proposed method has a potential for reducing human effort and transferring human skills to robots, which can be applied

to applications such as heavy load transport in automotive industry and compliant machining.

REFERENCES

- [1] P. Tsarouchi, S. Makris, and G. Chryssolouris, "Human-robot interaction review and challenges on task planning and programming," *International Journal of Computer Integrated Manufacturing*, vol. 29, no. 8, pp. 916–931, 2016.
- [2] J. D. Brown, C. E. O'Brien, S. C. Leung, K. R. Dumon, D. I. Lee, and K. J. Kuchenbecker, "Using contact forces and robot arm accelerations to automatically rate surgeon skill at peg transfer," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 9, pp. 2263–2275, 2017.
- [3] B. Jakopin, M. Mihelj, and M. Munih, "An unobtrusive measurement method for assessing physiological response in physical human-robot interaction," *IEEE Transactions on Human-Machine Systems*, vol. 47, no. 4, pp. 474–485, 2017.
- [4] H. Yu, S. Huang, G. Chen, Y. Pan, and G. Zhao, "Human-robot interaction control of rehabilitation robots with series elastic actuators," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1089–1100, 2015.
- [5] S. Cremer, K. D. Sumit, I. B. Wijayasinghe, D. O. Popa, and F. L. Lewis, "Model-free online neuroadaptive controller with intent estimation for physical human-robot interaction," *IEEE Transactions on Robotics*, vol. 36, no. 1, pp. 240–253, 2020.
- [6] Y. Li and S. S. Ge, "Human-robot collaboration based on motion intention estimation," *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 3, pp. 1007–1014, 2013.
- [7] D. A. Bristow, M. Tharayil, and A. G. Alleyne, "A survey of iterative learning control," *IEEE Control Systems*, vol. 26, no. 3, pp. 96–114, 2006.
- [8] T. Q. Ngo, Y. N. Wang, T. L. Mai, J. Ge, and T. L. Mai, "An adaptive iterative learning control for robot manipulators in task space," *International Journal of Computers Communications and Control*, vol. 7, no. 3, pp. 518–529, 2014.
- [9] F. Bouakrif, D. Boukhetala, and F. Boudjema, "Velocity observer-based iterative learning control for robot manipulators," *International Journal of Systems Science*, vol. 44, no. 2, pp. 214–222, 2013.
- [10] F. Bouakrif and M. Zasadzinski, "High order iterative learning control to solve the trajectory tracking problem for robot manipulators using lyapunov theory," *Transactions of the Institute of Measurement and Control*, pp. 1423–1435, 2018.
- [11] H. S. Ahn, Y. Q. Chen, and K. L. Moore, "Iterative learning control: Brief survey and categorization," *IEEE Transactions on Systems Man and Cybernetics Part C*, vol. 37, no. 6, pp. 1099–1121, 2007.
- [12] D. Shen, "Iterative learning control with passive incomplete information," *IEEE/CAA Journal of Automatica Sinica*, vol. 5, no. 5, pp. 885–901, 2018.
- [13] B. Ren, X. Luo, and J. Chen, "Single leg gait tracking of lower limb exoskeleton based on adaptive iterative learning control," *Applied sciences*, vol. 9, no. 11, pp. 2251–2260, 2019.
- [14] X. Li, J.-X. Xu, and D. Huang, "Iterative learning control for nonlinear dynamic systems with randomly varying trial lengths," *International Journal of Adaptive Control and Signal Processing*, vol. 29, no. 11, pp. 1341–1353, 2015.
- [15] L. Wang, X. Li, and D. Shen, "Sampled-data iterative learning control for continuous-time nonlinear systems with iteration-varying lengths," *International Journal of Robust and Nonlinear Control*, vol. 28, no. 8, pp. 3073–3091, 2018.
- [16] C. Liu, D. Shen, and J. Wang, "Adaptive learning control for general nonlinear systems with nonuniform trial lengths, initial state deviation, and unknown control direction," *International Journal of Robust and Nonlinear Control*, 2019.
- [17] X. Li, J. Xu, and D. Huang, "An iterative learning control approach for linear systems with randomly varying trial lengths," *IEEE Transactions on Automatic Control*, vol. 59, no. 7, pp. 1954–1960, 2014.
- [18] D. Shen and J. Xu, "Adaptive learning control for nonlinear systems with randomly varying iteration lengths," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 4, pp. 1119–1132, 2019.
- [19] H. Seraji and R. Colbaugh, "Force tracking in impedance control," *International Journal of Robotic Research-IJRR*, vol. 16, no. 2, pp. 499–506, 1993.
- [20] A. G. Feldman, "Once more on the equilibrium-point hypothesis (λ model) for motor control," *Journal of Motor Behavior*, vol. 18, no. 1, pp. 17–54, 1986.